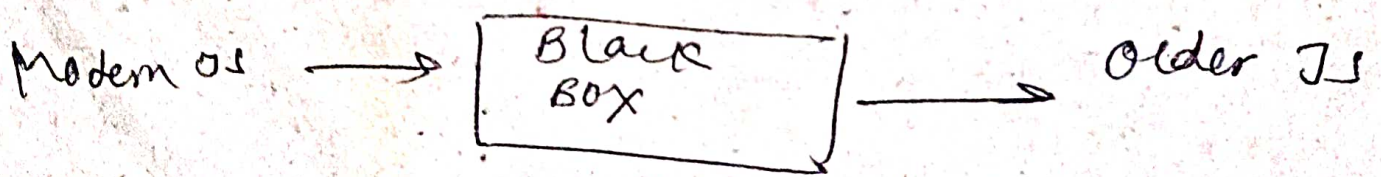


Laying the Foundation

- experiment with create React App.
- `Array.map()` → `function mymap()`
`polyfill`
- `Babel` does all these creation of `polyfill` to make our website compatible with older browser as well. `Browserlist`

→ Babel is a transpiler or compiler



→ git unit > to make an projects compatible with version control.

→ Parcel also does Tree shaking (removes unwanted codes)

Illustration > when we uses some library we doesn't use all the functions, so its make are to include only those function in production by build which has been used by an App (Project)

or Now one can build performant web scalable application?

→ npx parcel as well as index.html can be written to ^ package. For as

```
"scripts" : {  
  "start" : "parcel index.html",  
  "build" : "parcel build index.html",  
}
```

→ Now we can use command npm start to start our project.

→ Babel and Parcel doesn't need to configure log it directly. we need to

• babel-plugin-transform-remove-console

one thing to notice while installing package

→ whether you want to install as normal dependency or not.

→ Installing package doesn't mean we will be able to use it, we need to configure it as well.

• babelrc

→ Whenever there are siblings of a parent element multiple children of a parent element we need to give keys to the children element.

→ Key is anything which uniquely identifies an element.

→ Why keys? (React Reconciliation)

→ render } updating something on the DOM.

→ React uses the key to match children in the original tree with children in the subsequent tree.

→ Keys are passed as props to the ReactElement.

Introducing JSX

→ React.createElement is a very tedious and cumbersome method to write HTML like code inside React.

JSX comes to

the party.

React.createElement → object → HTML DOM

→ There is no any official full name of JSX but in development people like to call it as ^{typed} JavaScript XML

→ Major theory of bringing React is to update our ^{whole} HTML using JavaScript (because JS is a web ^{performant} app).

JSX example

```
const heading2 = (
  <h2 id="title" key="h.4">
    Namaste React
  </h2>
)
```

Q Is JSX, HTML inside JavaScript?
Ans → No it's not like syntax

→ This is JavaScript. is camel casing

→ In JSX, everything in JS
came as ~~class~~ JSX

HTML

class = "root"

className = "root"

Q How does JSX execute? Babel
Ans It is compiled by (or compiler),
only (a transpiler)

→ JSX uses the React.createElement
behind the scenes, done by Babel
JSX ⇒ React.createElement ⇒ Object
HTML DOM
a JavaScript compiler.

Babel is the advantages of JSX?

Q What are the advantages of JSX?
Ans 1) Readability friendly
2) developer friendly
3) syntactic sugar

→ Babel comes along with parcel (if
we install parcel, Babel also comes
along with it).

React.createElement over

React Component starts ↑↑
is a component in

→ Everything is a component in React.
→ Two types of components are there
- Functional } new way (recommended)
- Class Based components } older way of writing code
(learn it for interviews).

→ Functional component is nothing but a JavaScript function which returns a React element or composition element, or some piece of JSX, or component itself.

It is good practice to write start the name of functional component with capital letter.

e.g. `const component = () => {`
 `return <piece of JSX>`

`};` (You can skip return as well).
(Arrow function) [functionality].

↓
the way of writing it in production:

→ Rendering React element
`root.render(<React element name>);`

→ Rendering React component
`root.render(<Fxn component />);`

as how to use React element
inside functional component
↓ use it like
 `<React element />` → putting it inside curly
 → a way to write JSX inside
 JSX.

Q1 How to use Functional component inside Functional component

Ans Exn Component = $\langle \rangle \Rightarrow \textcircled{\langle \rangle}$

Way 1) < Functional components >
or

Way 2) < Functional component / > Good way

It is just basic JavaScript.

→ Any piece of JavaScript can be written inside JSX using curly braces.

→ $\langle \{ \} \rangle$ will be executed on the server.

→ XSS (Cross Site Scripting Attack).

It will be sanitized so no any attack will occur on JS code inside JSX.

→ Component composition (use of component inside another component)

(Nested component use),
the name for passing components as props to other components, thus creating new components with other components.

→ learn about Reconciliation.

Q1 Is the component hoisted?
Ans1 Functional component will behave
same as javascript function.

→ Create React App already comes
with webpack and better
so, that's why we don't configure
each stuff while doing Create React
App.